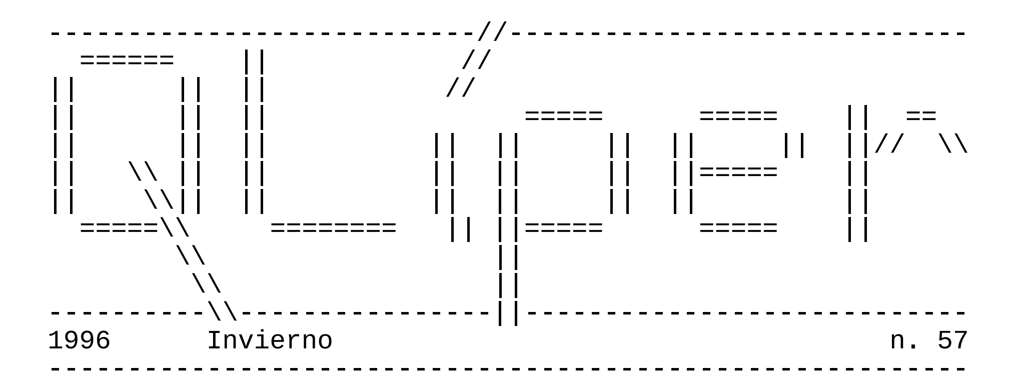


1996 Invierno n. 57

Estamos en el año 16 Después de Sinclair. Toda Hispania está ocupada por las legiones de PCs. ¿Toda? ¡No! Un puñado de irreductibles QLs resiste todavía y siempre ante el primitivo invasor...

QLíper
Redactor: Salvador Merino
Tel. (95) 2475043 (Trabajo)
(95) 2474887 (Casa)
Cerámicas Mary
Ctra. Cádiz (Torreblanca del Sol)
29640 FUENGIROLA (MALAGA)
ESPAÑA

INFORMACION SOBRE QLIPER

Compilación de colaboraciones y distribución: Salvador Merino
Portada: Pedro Reina

SE CONSIENTE LA REPRODUCCION TOTAL O PARCIAL DEL CONTENIDO DEL FANZINE, PARA USO CULTURAL Y NO COMERCIAL, POR CUALQUIER MEDIO FISICO, QUIMICO, OPTICO, MAGNETICO, SOLAR, MECANICO, HIDRAULICO, EOLICO, ELECTRICO, NUCLEAR O A PEDALES, SIEMPRE Y CUANDO SE CITE LA PROCEDENCIA (¿EH? ¡OJO!).
El fanzine QLIPER es un medio de comunicación e intercambio de ideas y conocimientos entre usuarios de ordenadores Sinclair QL y compatibles. El fanzine se realiza mediante la colaboración desinteresada de todas las personas que lo desean.

Indice

- Editorial

- Noticias

  Las noticias de Otoño'95

- Cartas Abiertas

  Carta de Marcos Cruz

- Respuestas

  Respuestas a la mayoria de las preguntas y dudas de Marcos Cruz

- QXL

La pantalla en la QXL
La QXL y Yo

- FORTH

SuperFORTH v2.0

- Z88

En busca del Z88 perdido

- Otros temas

Historias para no dormir por culpa de mi PC

- Programas contenidos en este disco

Editorial

Comienza el año 11 de vida de nuestro club, y el número de miembros/socios se ha estabilizado en el número perfecto de DIEZ.

Una buena noticia: Marcos Cruz ha dado señales de vida después de casi 2 años largos ausencia en las páginas de QLIPER. Tengo que advertir que una de mis principales razones por las cuales estaba bastante desmoralizado en los últimos tiempos, era porque me habia quedado sin la ayuda o cooperación de los socios/miembros fundadores de la segunda etapa del CLUB de USUARIOS QL ESPAÑOL. Todo esto último significaba el fin del Club, pues yo ya estoy quemado, agotado y sin el entusiasmo de los años ochenta. Sin embargo, gracias a las colaboraciones de Pedro Reina, el club continua vivo.

La pregunta que más hacian los usuarios de QXL últimamente era: ¿Cómo acceder directamente a la pantalla QXL en resoluciones EGA, VGA y SVGA?. En esta revista vais a encontrar un articulo en el cual he intentado explicar con todo detalle mis experiencias, pues como ya sabeis, el manual de la QXL deja mucho que desear, y no he adquirido aún documentación sobre el sistema operativo SMSQ. Sin embargo, yo creo que con mis modestos apuntes, cualquiera puede escribir sus rutinas para acceder a la memoria de pantalla en las diferentes resoluciones, e incluso hacer volcados gráficos de ventanas/consolas a una impresora.

Sobre el futuro del mundo QL, parece que tenemos una suerte negra, pues la Comunidad Europea sin saberlo nos ha torpedeado por debajo de la linea de flotación. En otras palabras, la normativa EMC EC, que entra en vigor el 1 de enero de 1996, deja al viejo SINCLAIR QL sin posibilidad de obtener nuevo hardware de actualización. Afortunadamente, el mundo QL tiene varias alternativas para continuar viviendo sin depender de la placa madre QL original.

En el pasado número QLIPER se informaba de la existencia de un emulador QL para Apple MAC y PowerMAC escrito en 'C'. Ahora puedo anunciar que un emulador QL llamado QPC para PC 486/PENTIUM ha sido mostrado en la feria QL Alemania de septiembre'95. La verdad es que QPC no es rival para una QXL (siempre y cuando no corra en un PC PENTIUM PRO a 300 Mhz), pero es una alternativa barata para correr software QL en un PC -sin utilizar un destornillador-.

No creo que tenga serios problemas para llenar los discos QLIPER de 1996, pero ruego a todos los socios que aporten su granito de arena en la compilación de la revista (la cuota de socio no da derecho a exigir nada, pues solamente cubre unos gastos minimos. Nuestra revista es solamente un medio de intercambio de información entre los socios del Club).

Os deseo una feliz navidad, y que todos vuestros proyectos o sueños se hagan realidad durante el año 1996.

Salvador Merino, 10/11/1995

Noticias

LAS NOTICIAS DE OTOÑO'95

Pocas novedades hay en el mundo QL, pero las novedades del Otoño'95 se pueden resumir en los siguientes puntos:

- PRIMUS 68K INDUSTRIAL COMPUTER.- Es un QL compatible para uso industrial que se compone de:

1.- Micro computer.- Una placa de 19 pulgadas (220x100 mm) basada en el 68000 a 16 MHz y 2 Mbytes de RAM estatica. Sistema operativo MINERVA.
2.- Tarjeta compatible QL.- Es una tarjeta de 160x100 mm que contiene componentes QL especificos como ZX8301, ZX8302, IPC 8049 más 32 Kbytes de memoria pantalla.
3.- Tarjeta con 7 ranuras de expansión.
4.- Tarjetas expansión.- Se pueden conectar tarjetas de expansión QL.
5.- Caja de 19 pulgadas.- La caja tiene una fuente de alimentación de 93V a 264V.

Más información:

Ing. Büro Michael Klein
Martinstrasse 23
64285 Darmstadt
Alemania

- QPC.- Es un programa para PC que emula el Sinclair QL. En un PC 486 a 33 MHz alcanza la velocidad del QL original. Si la QXL no vuelve a fabricarse, QPC podria ser nuestro bote salvavidas, pero necesitariamos un PC PENTIUM PRO a 300 MHz para alcanzar la velocidad de una QXL 68040 -sin QL NET-.
Como dato curioso, el programador de QPC es un adolescente que era un recien nacido cuando el QL nació.

- Pantalla LCD para QL.- Zeljko Nastasic ha mostrado en la feria QL en munich (22-24 sept'95) una pantalla LCD monocroma controlada por un QL. Parece que es el primer paso para el desarrollo de un QL portátil.

En resumen, diez años después de la muerte oficial del QL, todavía continua el desarrollo del proyecto QL.

Salvador Merino, 20/11/1995

Cartas Abiertas

Carta de Marcos Cruz

Madrid, agosto-octubre de 1995

Salud, discípulos de Sinclair, sí, soy Lázaro, digo Marcos. Como probablemente ya os habréis olvidado de mí, quiero contaros qué ha pasado en todo este tiempo, y por qué he estado tan ausente.

Sólo algunos de vosotros habéis tenido noticias mías en este tiempo: Salvador, la última vez hace unos meses cuando le llamé para decirle que tenía una QXL (desde entonces no he dado señales de vida, lo siento); Pedro Reina, a quien le he consultado algunas cosas sobre la QXL cuando el horizonte se me ponía muy negro; y Joaquín Gallardo, a quien le pedí

información sobre su Z88, que es español cuando yo pensaba que era inglés.

Bueno, la historia es la siguiente: Durante muchos meses tuve mi QL con mi vieja Trump Card totalmente infrautilizado, porque las unidades de disco fallaban muy a menudo y era imposible hacer nada, hasta que dejaron de funcionar y entonces sí que fue más imposible si cabe. Durante un tiempo no me podía pagar unas unidades nuevas, hasta el punto de que estuve dándole vueltas a la idea de utilizar las del PC, por medio de un selector de esos que se usan para conectar varias impresoras, pero no llegué a cojer el destornillador.

Hace un año me surgió una posibilidad de trabajo en una productora de publicidad. Colaboré ocupándome de los recién llegados Mac, más que nada aprendiendo, porque nunca había visto un Mac sino en foto. Rápidamente me hice con ellos, con los problemillas de la red y con los programas integrados que se usaban diariamente, de modo que pude a su vez ayudar a manejarlos más eficazmente. Después me metí con el File Maker Pro, un creador de bases de datos bastante original y cómodo. No es programable ni relacional, por lo cual tiene algunas limitaciones, pero es muy amable, rápido e intuitivo, y permite obtener resultados bastante aparentes en poco tiempo a base de macros, botones y demás. Así pues, hasta finales de año me ocupé de crear con FMP un sistema de gestión de empresas y personas a medida de las necesidades de la actividad, así como otras bases de datos y modelos para presupuestos, facturas, etc. Por todo esta historia pedí una gratificación de 100.000 pesetas, que en principio pensaba ahorrar para seguir pagando mi impagado PC. Ocurrió que la empresa se iba a dividir, y mi situación en ella, o mejor dicho en una de las dos futuras mitades, iba a regularizarse, y tendría un sueldete de 100.000 pesetazas desde diciembre, de modo que decidí invertir mis primeros honorarios en mi olvidado QL. Ante la duda de si comprar una nueva unidad de disco y una Super Gold Card, o bien una QXL, me decidí por esta última opción y compré una QXL de 4 Mo. Pensé que era una inversión mejor, ya que de esa manera no duplicaba monitores ni unidades de disco ni tendría que andar de un lado a otro con la impresora. Además, tendría el disco duro del PC para usarlo con el QL.

La casualidad hizo que en ese mismo momento recibiera una tarjeta de DP con la oferta de una colección de sus mejores programas por 99 libras. Ya desde hacía tiempo pensaba comprar el SuperForth, así que me lanzé y pedí la cole completa, aunque sabía que la mayoría de los programas no los iba a usar nunca, ya que la toda mi actividad personal (textos, bases de datos, hojas de cálculo) la realizo con el PC. Quizá me decidió el pensar que de esa manera podría usar el QL para todo lo que quisiera, si alguna vez me hacía falta.

En suma, que en cosa de un mes estuve reequipado para la vida moderna. Por desgracia, el trabajo se hizo cada vez más absorvente e imprevisible, y apenas tuve tiempo ni fuerzas para nada más durante los cinco meses siguientes.

A finales de abril ya no pude más y abandoné la productora. La cosa se puso muy seria. Agotado, estresado y con cinco kilos menos, me di cuenta (y no era la primera vez) de que la publicidad no era lo mío, por muy condimentada de informática que estuviera.

Aparte de conocer el mundillo de la producción de cine publicitario en general, lo más fructífero que he sacado es la experiencia con los Mac y los Power Mac. Esto me ha servido, de paso, para relacionarme con más objetividad con el entorno QL al regresar a él después de tan largo exilio. Por un lado ha sido un placer volver a encontrar cosas que son mucho más cómodas de hacer en el QL que en Mac o PC; por otro lado, he echado de menos otras que el QL no tiene, o aún no han sido desarrolladas.

Bueno, después de un par de meses de zozobra, he cogido un trabajillo de reponedor en Pryca, de cuatro horas diarias

a 500 pesetas la hora, para ir tirando. Gano la mitad que antes pero parece que poco a poco estoy volviendo a mi peso normal. Y os aseguro que es más entretenido que lo otro, aunque no lo creáis. Y para colmo tengo tiempo para vivir, así que qué más se puede pedir de momento... aparte de una tarjeta gráfica para la QXL con resolución SVGA y compatibilidad con los programas existentes, claro (y de paso, si no es mucho pedir, juegos de caracteres escalables, o al menos más grandecitos).

Bueno, pues esa es la historia. Empecé este último trabajo a finales de junio, y después de un mes de adaptación a nuevos horarios y ritmos vitales (entro a las 6:30 de la mañana), he empezado de nuevo a ponerle las manos encima al invento este del señor Sinclair.

Por cierto, antes de encontrar este último trabajo instalé una línea nueva de teléfono para poner anuncios para dar clases particulares y hacer otro tipo de encargos computeriles. Me podéis localizar pues en el 91-3513372. Cuando estoy durmiendo o no estoy en casa, tengo un contestador (que a veces está mudo, pero funciona).

Salvador Merino nunca se da por vencido, y pese a que llevo prometiéndole colaboraciones desde hace un año, y después de contarle que tenía una QXL y guardar silencio absoluto durante meses, aún me envió los últimos números de QLíper, e incluso una tarjeta postal anunciándome la aparición de un emulador de QL para Mac... gracias, Salvador, no te merecemos.

Os mando unas colaboraciones:

-forth_doc sobre mis infructuosos experimentos con SuperForth 2.0 (especial para Salvador, que debe de ser el último Forth-adicto que queda).

-QXL_doc sobre la susodicha y genial tarjeta.

-Z88_doc o en busca del Z88 perdido.

-Spectator 1.85, emulador de Spectrum para QDOS, que funciona muy bien en la QXL. Se necesita una Gold Card para que la velocidad sea aceptable. El la versión completa, pero es pospago ("shareware") y puede distribuirse gratis. La pedí del autor, Carlo Delhez, al ver que la versión antigua que yo tenía no funcionaba en la QXL. Puede que me registre.

-Z80 3.03, la última versión de un buenísimo emulador de Spectrum para PC. El fichero comprimido contiene la versión pospago, que es totalmente operativa, salvo que no se puede reducir la velocidad y que carece de ficheros de utilidades adicionales. De este programa soy usuario registrado desde hace varias versiones. Es una maravilla. Hasta he construido una pequeña interfaz (¡dentro de una lata de atún!), siguiendo las indicaciones del manual (lo de la lata de atún no lo dice el manual), para leer cintas del Spectrum a través del conector de impresora del PC, y funciona a la perfección. Ya he pasado cerca de 8 megaoctetos de programas del Spectrum al formato .TAP y .Z80 de este emulador (compatibles con el Spectator, por cierto).

-Ficheros de mi disco duro, relacionados con la organización y funcionamiento de mi QXL, por si a alguien le sirven de algo. Aparecen explicados en qxl_doc.

¡Socorro!

1) ¿Alguien me puede proporcionar el formato interno de los ficheros DBF? ¿Es igual en dBase III y en dBase IV? Me interesa el primero. Además, me gustaría conocer también el formato de los correspondientes NDX (y si no es mucho pedir, de los equivalentes NTX de Clipper). Del formato DBF sé que lo único interesante es el comienzo del fichero, donde se guarda la definición de los campos, pues creo que después todos los campos van uno detrás de otro sin más misterio. Me gustaría tenerlo para utilizar ficheros de dBase desde Forth, o desde SBASIC.

2) ¿Alguien sabe algo de un programa llamado Quickfax, escrito por Simon Goodwin, y anunciado como "proyecto aún sin terminar" en un anuncio de Creative Works en el número de mayo de 1988 de SQLW? Se trata de un sistema de base de datos. ¿Alguien sabe si llegó a publicarse alguna vez, o sabe algo más de él. Por la descripción que he leído en dicho número de la revista, parecía muy prometedor.

3) Tengo una copia en cartucho del Chess, que me vino con el penúltimo QL de segunda mano que compré. Aparte de que de vez en cuando me da algún problema al cargar, quisiera saber cómo pasarlo a disco. Lo intenté hace tiempo de varias maneras, pero no funcionaba. ¿Alguien sabe algo?

QL frente a Mac y PC

Como os comenté antes, el trabajar habitualmente con PC desde hace tiempo y el haber adquirido últimamente un buen conocimiento y experiencia de los Mac ha hecho que la "vuelta" al mundo del QL me haya provocado impresiones contradictorias, que resumo de la siguienta manera:

Puntos fuertes del QL, con los que ha sido agradable volverse a encontrar después de tanto tiempo:

-Multitarea sencilla, cómoda y eficaz.
-Fácil de programar para cualquier cosa.
-Programas de pequeño tamaño.

Puntos débiles del QL, o aspectos en los que he echado de menos a los otros sistemas:

-Interfaz de usuario anticuada.
-Interfaz gráfica (PE) poco extendida y compleja de instalar y configurar.
-Los programas son cada uno de su padre y de su madre, principalmente en lo que respecta a la interfaz con el usuario, al manejo de dispositivos de almacenamiento y de la impresora (algo así como sucede en los programas de MS-DOS en relación a los programas de Windows y MacOS).
-Pocas posibilidades para sacar el máximo partido de cualquier impresora, ¡incluso en los programas de tratamiento de textos! Esto se debe a la falta de estandarización y de controladores universales que funcionen con todos los programas (vale la misma comparación anterior de MS-DOS con Windows y MacOS).

En resumen, el QL sigue compensando para ciertas cosas, pero hay que hacer evolucionar ciertos aspectos si no queremos que se quede estancado en métodos de funcionamiento propios de hace 10 años. Es cierto que el corazón del QDOS es muy avanzado, pero su interfaz no se diferencia mucho de la de MS-DOS, al menos en lo fundamental.

En mi opinión, hay tres caminos por los que hacer evolucionar el sistema:

1) Equipos rápidos y potentes como la QXL, que permitan actualizaciones del sistema operativo, así como utilizar interfaces y dispositivos estándar del mercado. En mi opinión el camino ha de ser la QXL, no la SGC, pues es un camino de apertura que nos pone en contacto con el mundo real, codo con codo con los otros sistemas.

2) Sistemas operativos compatibles con QDOS y más potentes, así como actualizables sin necesidad de cambiar de equipo, como SMSQ y otros.

3) Desarrollo del PE, desarrollo de nuevos programas para él y adaptación de programas ya existentes.

Un aspecto fundamental que le falta al mundo QDOS, y por desgracia imprescindible para ser tomado en serio fuera de su

reducido mundo, es la resolución gráfica. Esto entra dentro del punto 1 mencionado anteriormente. No sé hasta qué punto es posible conseguir una resolución VGA (mínima necesaria actualmente para salir al "exterior") y al mismo tiempo conservar la compatibilidad con programas pasados y futuros. Creo que Miracle ya ha sacado su tarjeta gráfica, pero no sé nada de ella. Desde luego, con la resolución y juegos de caracteres actuales no creo que nadie se tome muy en serio el QDOS fuera de "nuestras fronteras"; por desgracia muchas cosas entran por los ojos... Windows mismamente entra por los ojos, aunque interiormente sea una patata. Hay que ser consciente de ello. El aspecto de muchos programas para QL, basados en menús y en extraños comandos que hay que memorizar, no pasaría un examen serio. Por eso creo que el PE ha de ser el camino, pues es la única vía que tenemos para proporcionar uniformidad y comodidad al sistema.

Respuestas

Respuestas a la mayoria de las dudas y preguntas de Marcos Cruz

En la presente revista QLIPER Marcos Cruz vuelve a llenar sus páginas, pero como cualquier mortal interesado en aprender, hace muchisimas preguntas. Mis respuestas a la mayoria de sus preguntas son:

- La información sobre los ficheros DBF y NTX la tiene Pedro Reina, y puedes medio entenderla en los ficheros fuente del sistema OLIMPO 2.0. Mi lenguaje MERINO dBASE TIL v3.0 ya tiene todas las palabras del sistema OLIMPO v2.0.

- El programa Quickfax de Simon Goodwin me temo que nunca ha sido comercializado, pero si sé que habia algunas copias betas corriendo por ahi.

- El CHESS del CST THOR corre muy bien con la QXL y los QLs ampliados.

- Tu problema con SuperFORTH fue el primero que tuve yo en 1987 cuando deseaba crear mi primera aplicación Stand-Alone. Debes utilizar la palabra OPEN_DEVICE en vez de OPEN, pues la última solamente funciona en modo intérprete, y la primera en ambos (intérprete y compilado). El problema de OPEN es que abre o crea dispositivos en modo intérprete, pero cuando salvamos el sistema SuperFORTH ampliado o como una aplicación solitaria, esos dispositivos abiertos o creados se pierden. En otras palabras, hay que salvar el sistema de manera que una palabra compilada en el diccionario vuelva a abrir esos dispositivos, y eso solamente se puede hacer utilizando la palabra OPEN_DEVICE.

- El fichero conteniendo el fuente del programa REVERSI no venia en la versión comercial del SuperFORTH, solamente el listado en papel de una versión antigua de REVERSI para FORTH79 (el antecesor de SuperFORTH), la cual no tiene el código preparado para convertirla en una aplicación solitaria.

- Dudas sobre la pantalla en la QXL:

1) Saber en qué resolución gráfica estamos trabajando.- Para conocer la resolución máxima de una ventana en 'C68' o ASM solamente tenemos que redefinir un canal de ventana ya abierto (si da error es que no permite esa resolución).

2) Saber la dirección de comienzo de la pantalla.- Eso es fácil, en el bloque de definición de una ventana (canal de la ventana) *SD.SCRB $32 contiene la dirección base de la pantalla (en un QL era siempre la misma dirección, pero en una QXL u otros sistemas QL compatibles puede variar).

3) ¿Es Modo 4 o 8?.- Se puede obtener con el TRAP#1 MT.DMODE.
4) La forma de guardar los colores en las diferentes resoluciones es la misma que en el QL original, solamente cambia en ancho y alto en pixels.

- La QXL puede utilizar cualquier sistema de almacenamiento en disco. Por ejemplo: C: es WIN1_, D: es WIN2_,... Se puede crear un disco duro en un fichero QXL.WIN y grabarlo en un disco CD-ROM para utilizarlo como WIN2_ en una unidad lectora CD-ROM en D:.

- Para configurar el ratón en SER1 en mi PC, hay que utilizar el programa CONFIG del QPTR (o QPAC II). La configuración es: SER1 > none, y SER2 > COM2/IRQ3. Para que la configuración sea un éxito, hay que resetear o apagar/encender el PC y volver a ejecutar el nuevo SMSQ.EXE configurado.
Si he de ser sincero, yo habia configurado mi SMSQ.EXE el año pasado sin éxito (lo habia dejado por imposible sin saber el por qué), porque probaba el ratón en el QPTR sin volver a resetear o reiniciar (apagar/encender) mi PC. Ha sido un capricho del destino el que me diera un día por mover el ratón y comprobara que por arte de magia yo habia configurado correctamente el ratón, pero ni me habia enterado de que lo habia hecho.

- Cambiar la Eprom de un Z88 Inglés por una Español no es suficiente. También seria necesario cambiar el teclado. La ROM es de 128 Kbytes.

- Todos los caracteres del Z88 están en el teclado. El Z88 Español posee algunos extras en Rojo. Me temo que en un Z88 Inglés es imposible acceder a caracteres Españoles, pero tiene la ventaja de poder utilizar un paquete de comandos gráficos para el BASIC que solamente funciona en las primeras versiones Inglesas de la ROM.

- El Z88 solamente sirve para utilizarlo como procesador de texto y hoja de cálculo principalmente. Otros usos podrian ser conectarse con periféricos utilizando su interface RS-232. Sin embargo, hoy en día el Z88 está anticuado simplemente porque el microprocesador Z80 es ya historia, el sistema de almacenamiento en EPROM sigue siendo caro, la RAM consume muchas pilas,...

Salvador Merino, 23/10/1995

QXL

LA PANTALLA EN LA QXL
=====================

En el QL original solamente habia un mapa de pantalla que comenzaba en la dirección 131072 y ocupaba 32 Kbytes. Sin embargo, actualmente hay sistemas QL compatibles que disponen de mayores resoluciones gráficas, y el problema es: ¿Dónde está el mapa de la pantalla?. Afortunadamente, todo estaba previsto en el sistema operativo QDOS original, y no ha sido dificil dar solución al problema.

En la tabla que define un canal SCR o CON abierto hay una constante o variable de 32 bits llamada SD.SCRB ($32) que el sistema operativo llena con la dirección de comienzo del mapa de la pantalla.

¿Cómo acceder a la tabla de definición de un canal SCR o CON?

Afortunadamente, existe un TRAP llamado SD.EXTOP que permite añadir funciones a los drivers de pantalla. En el libro "QL programación Avanzada" está tan mal explicado que uno no sabe ni para qué sirve, pero en el libro "QDOS COMPANION"

se explica con todo detalle. El registro A2 debe contener la dirección de comienzo de una rutina, y lo bueno es que cuando se ejecuta esa rutina el registro A0 contiene el comienzo de la tabla de definición del canal.

El siguiente programita en C68 consta de dos listados, uno en 'C' y el otro en assembler, que hay que compilar y linkar. El programa resultante lo único que hace es mostrar en pantalla la dirección del mapa de pantalla actual.

```
#include <olimpo.h>

void rut_direc_pant();

main()
{
long canal;
long *bufa1;
long direccion_canal;
char buffer[4];
canal = fgetchid(stdout);
sd_extop((fgetchid(stdout)), -1L, rut_direc_pant, 0, 0, bufa1);
direccion_canal = *bufa1;
printf("La dirección del mapa pantalla es %d\n", direccion_canal);
getch();
}
```

```
.globl     rut_direc_pant

rut_direc_pant:
                move.l      50(a0),(a1)
                moveq       #0,d0
                rts
```

Si la dirección del mapa de pantalla no es 131072, solamente significa una cosilla: nos encontramos con una resolución diferente a la del QL original. Aunque el operador ya lo sabia, nuestro programa no lo sabia.

¿Cómo adivinar la resolución gráfica en cualquier sistema QL?

Solamente se me ocurre una idea. Consiste en que una vez comprobado que el mapa de la pantalla no corresponde al de un QL original, pasemos a utilizar una pequeña tabla de resoluciones conocidas de otros sistemas QL compatibles. En otras palabras, vamos a utilizar el TRAP SD.WDEF para comprobar una tabla de resoluciones de mayor a menor, lo que significa que la primera que no de error será la resolución de la pantalla.

Una QXL dispone de las siguientes resoluciones gráficas:

	MODO 4	MODO 8	Longitud
QL	512x256 pixels	256x256 pixels	32768 bytes
EGA	640x350 "	320x350 "	56000 "
VGA	640x480 "	320x480 "	76000 "
SVGA	800x600 "	400x600 "	120000 "

Para distinguir los ficheros de pantallas completas habria que abrirlos con el TRAP IO.OPEN y leer sus cabeceras de fichero con el TRAP FS.HEADR para conocer su longitud. Una vez conocida su longitud, sabriamos a que resolución pertenece. Los ficheros de pantallas de resoluciones inferiores se podrian leer para una resolución superior, pero habria que leerlos por lineas y escribirlos en la memoria de pantalla por lineas.

¿Cómo saber en qué modo estamos trabajando?

Fácil, el TRAP MT.DMODE permite ajustar o leer el modo pantalla (4 o 8 colores).

Los cálculos para hallar la posición de memoria de cualquier pixel en cualquier resolución son:

```
x1 = ancho_pixels/4;  /*  el ancho en pixels puede ser 512, 640 o 800 */

/*   MODO 8  */

byte_par   = direccion_pantalla + y*x1 + (x/8)*2;
byte_impar = direccion_pantalla + y*x1 + (x/8)*2 + 1;
posicion_bit_dentro_de_byte = 7-(x%8);

/*    MODO 4  */

byte_par   = direccion_pantalla + y*x1 + (x/4)*2;
byte_impar = direccion_pantalla + y*x1 + (x/4)*2 + 1;
posicion_bit_verde_rojo    = 7-(x%4);
posicion_bit_azul_parpadeo = 7-(x%4)+1;
```

En resumen, espero que todo lo anterior sirva de ayuda para aquellos que deseen utilizar las resoluciones EGA, VGA y SVGA de la QXL, y no sabian cómo. Todos los TRAP mencionados en este articulo están disponibles en C68, pero para SBASIC habria que escribir como minimo una función en assembler para conocer la dirección de la pantalla y otra función redefina una pantalla retornando 0 si todo ha salido bien o un error.

Salvador Merino, 8/11/1995

LA QXL Y YO

1. Primer contacto con la QXL

Lo primero que me llamó la atención al poner en marcha la QXL, por lo novedoso para mí, fue ver la pantalla del QL en color, pues sólo alguna vez hace años había usado el QL con una televisión. Siempre lo he tenido con un monitor de fósforo verde. Ya veis qué tontería, me quedé un poco atontado ante tanto colorido... casi no hay palabras para describir una gama tan amplia de colores y tonalidades. Además, el monitor que tengo en el PC es de buena calidad, así que se ve de maravilla; nunca había visto la pantalla del QL tan nítida.

Una de las primeras cosas que hice con la QXL es conectarla en red al QL con Trump Card y probar FSERVE, pues nunca había podido hacerlo, ya que mis otros tres QLs no tienen ampliación de ningún tipo. Me quedé maravillado de lo fácil, cómoda, sencilla y eficaz que es la comunicación. Cualquier dispositivo de uno de los QLs podía estar disponible para el otro de la forma más natural. Increíble. Ya las redes de Mac son una gozada en comparación con las de los PCs, pero esto era demasiado bueno. No necesitaba configuración de ningún tipo, cosa que en los Mac hace falta y no es muy evidente ni intuitivo el proceso hasta que se le coge el truco. En el QL es casi totalmente transparente.

Otra cosa que me sorprendió muy gratamente es la posibilidad de tener varios BASIC al mismo tiempo. Es fantástico, incluso para el trabajo cotidiano normal y corriente.

Y por supuesto, aunque no haga falta mencionarlo, la velocidad me dejó muy impresionado. Hice algunas pruebecillas que luego os comentaré.

Un detalle muy cómodo es la capacidad de leer y escribir

directamente discos de formato MS-DOS. El manual menciona que se pueden hacer reconocibles más formatos "sobre la marcha", sin más explicación. ¿Qué quiere decir? O bien que en futuras versiones se podrán leer otros formatos, o bien que el sistema es ya ampliable y configurable en ese sentido. Como siempre, el manual de Miracle deja muchas cosas en el aire.

2. Problemas con la QXL

2.1. Puerto de PC.

Tuve que cambiar el puerto de entrada y salida del PC al que se conecta la tarjeta, porque en el 2B0 hex. original de vez en cuando el sistema salía al DOS con el mensaje "The QXl at IO address 02B0h is not responding". Empecé por probar el primero de la lista de alternativas del manual, el 280 hex., y de momento no ha vuelto a pasar.

2.2. Teclado.

El primer problema serio fue el teclado. Gracias a los programas de Salvador y Pedro Reina, aparecidos en uno de los últimos número de QLíper, comprendí cómo instalar una tabla de teclado a medida. Utilicé el programa de Pedro Reina y le hice algunas pequeñas modificaciones para acceder fácilmente a algunos caracteres adicionales que él no había incluido por no ser muy usuales. Después de esto pude por fin probar en serio la QXL. Mi versión del programa de Pedro Reina la encontraréis en este número de QLíper).

2.3. Incompatibilidad con programas antiguos.

Enseguida me di cuenta de que programas antiguos no funcionaban bien, o se bloqueaban. Los juegos de la colección de DP no pitan ni a rastras, aunque me trae sin cuidado.

Algunos hacen cosas raras, sobre todo con la pantalla. Spy, por ejemplo, se deja las ventanas del menú que sale con F3 en medio de la pantalla, como si se olvidara de borrarla, y hay que moverse en el texto o usar F4 para limpiar, aunque por lo demás va bien..

Ni Lightning normal ni la versión SE funcionan. No cuelgan el sistema, pero la impresión en pantalla se convierte en basura, como si las líneas de pixeles estuvieran descolocadas. El acelerador matemático de Lightning no acelera nada (más sobre esto después).

Hardback se autoelimina después de la primera pantalla.

2.4. Problemas con formateo de discos

Desde el primer día, y sin motivo aparente, los discos de alta densidad se formateaban como de doble densidad en la QXL, dando 1440 sectores. Para evitarlo, tenía que volver a formatearlos desde DOS como de alta densidad y repetir la operación. Aun así, a veces volvía a fallar. De hecho, pasaba incluso al volver a formatear discos que ya tenían 2880 sectores bajo la QXL; al volver a formatearlos, daban 1440. Incluí en el boot la instrucción FLP_DENSITY, que no viene en el manual, pero que leí en un Quanta; parece que no hizo nada, pues siguió pasando. Aún no he encontrado una explicación a esto, pero parece que pasa menos. Quizá fuera el verano.

TRA

Una de las cosas más primitivas del QL, en mi opinión, es el manejo de impresoras. Casi tan primitivo como en MS-DOS. en lugar de tener controladores universales para cada

modelo o tipo de impresora, cada programa se las apaña como puede, y el QDOS a su vez también, mediante traducciones de caracteres.

Esto hace que el esfuerzo de imprimir un texto en Xchange no sirva de nada para imprimirlo en Perfection o The Editor: en cada caso hay que empezar desde cero, configurando los controladores de turno y haciendo un sinfín de pruebas.

El comando TRA de la QXL es más avanzado que sus predecesores y permite utilizar tablas de traducción personalizadas. Se incluyen tres tablas a elegir: británica, alemana y francesa.

Lo primero que no entiendo es por qué no basta con una tabla única de traducciones, o al menos una por cada juego de caracteres posible en las impresoras (el IBM es el más común, pero puede haber ANSI, Latin 1, etc.). Quiero decir que el juego de caracteres del QL no cambia, no depende del idioma del usuario. Lo que cambia es el juego de caracteres de la impresora que estemos usando.

Gracias a la ayuda de Pedro Reina, que me facilitó el formato interno de las tablas de traducción, empecé a trabajar en un programa que me permitiera elegir la impresora con la que estaba trabajando y el modo de ésta, y cambiar el juego de traducciones en cualquier momento.

Se necesitaba pues reservar espacio en memoria para la tabla ya en el mismo Boot de la QXL, guardar la dirección de ese espacio en un fichero en el disco, para que no se perdiera, y por último un programa que presentara una lista de posibles impresoras y construyera y activara una nueva tabla de traducciones en esa misma dirección.

Después de una serie de fallos, producto de una serie de despistes míos, la cosa parecía que funcionaba, pero sólo para las traducciones simples, de uno a uno. Las traducciones de uno a tres no funcionan, y aún no he averiguado qué estoy haciendo mal. De momento estoy tirando con TRA 3, que sirve bastante bien, pues traduce el juego del QL al juego IBM de cualquier impresora que disponga de él (casi todas).

He dejado aparcado el asunto de TRA, a ver si cuando lo coja de nuevo, después de algún tiempo, lo veo más fríamente y descubro el fallo. Por si queréis curiosear, os envío los ficheros que lo componen (los que empiezan por S_TRA_).

Pruebas con la QXL

He hecho algunas pruebas y comparaciones con la QXL. No pretenden ser muy científicas ni rigurosas, sino sólo satisfacer cierta curiosidad, comprensible cuando se trabajaba con un equipo nuevo:

1. Lightning

En la colección de DP venía Lightning en su versión normal y SE. Aquélla ya la había comprado hace tiempo, y la usaba siempre en la Trump Card. Enseguida vi que ninguna de las dos funciona bien con la QXL. No obstante, quise probar la versión SE con la Trump Card (TC), y me llevé una gran sorpresa, al probar los dos testeos que viene con el paquete (las cifras están en segundos):

	test texto	test gráficos
TC	5369	82
TC + Lightning	3368	82
TC + Lightning SE	3390	121

O sea, que la versión SE no acelera nada en gráficos, y es algo más lenta en textos... ¡Increíble!. ¿Alguien ha hecho alguna compración similar entre los dos Lightning?

Mi única esperanza era que el acelerador de funciones

matemáticas hiciera algo en la QXL, así que escribrí un pequeño programa de testeo, pues no viene ninguno con Lightning. El programa lo podéis encontrar en este disco con el nombre marcos_TestMath_bas. Los resultados fueron más sorprendentes si cabe:

QXL	554 segundos
QXL + Lightning	813 segundos
QXL + Lightning SE	555 segundos

Si alguien me ayuda a interpretar todo esto, le estaré eternamente agradecido.

2. PC Conqueror

En la colección de DP vienen también las dos versiones de PC Conqueror (PCQ). Lamenté haber comprado en su momento la versión normal por un precio casi igual a lo que me había costado ahora la colección completa... Qué remedio. El caso es que quise hacer un test también con estos dos programas, por curiosidad. Para ello utilicé un programa de chequear PCs que tenía por ahí.

PCQ1 = PC Conqueror 1.16
PCQ2 = PC Conqueror Gold SE 3.02

Sistema principal:	PCQ1	PCQ2
UCP	8088	8088
MHz	4,20	4,20
I'ndice Dhrystones	2181	2144
Relación con PC XT	6,34	6,19
Velocidad matemática (sin NPU):		
I'ndice Whetsores	32,1 K	30,8 K
Relación con PC XT	4,86	4,66
Vel. de vídeo por BIOS (CGA):		
Caracteres/segundo	73	1054
Relación con PC XT	0,11	1,56
Vel. de vídeo directo (CGA):		
Caracteres/segundo	25635	26276
Relación con PC XT	5,13	5,26

Las pruebas están hechas sin modificar la configuración con que vienen los programas. Es soprendente que no haya casi diferencia entre las dos versiones, excepto en el acceso a vídeo por el BIOS, donde la diferencia es muy significativa.

3. QL80

Este programa es mi viejo ensamblador de Z80. En la Trump Card, ensamblaba un largo programa que escribo para Spectrum (desde hace 9 años...) en 20 minutos, desde disco. La misma operación tardó poco menos de la mitad en la Gold Card de Pedro Reina. En la QXL, ensamblando desde RAM1, tarda 1,39 minutos, a razón de 8340 líneas por minuto.

QXL + Windows

No sé por qué, pero nunca se me ocurrió probar esto, quizá porque suponía que era imposible. En un Quanta leí algo de un programa que permitía pasar de una tarea a otra de Windows más fácilmente que el propio Administrador de Windows, y que mediante él se podía ejecutar SMSQ desde Windows como si fuera un programa DOS cualquiera.
En el siguiente Quanta otro usuario de QXL decía que él

se había puesto a experimentar a raíz del artículo anterior y había logrado lo mismo sin programas especiales ni nada, y explicaba el procedimiento (el mismo que para cualquier programa DOS). Sorprendido, puse en marcha el PC, activé Windows, lo probé y funcionó a la primera. No podía creerlo: era como si el QL fuera un programa más bajo Windows (por supuesto a pantalla completa, pues como ventana no acepta teclas), con lo cual se podía simultanear con cualquier otro programa de PC.

Sólo encontré un pequeño problema: La combinación ALT INTRO está reservada por Windows para pasar de pantalla completa a ventana y viceversa, así que ese era su efecto en el SMSQ, lo cual era un engorro. Para solucionarlo, nada más fácil: en lugar de asociar el icono correspondiente al programa SMSQ.EXE, hay que crear un fichero SMSQ.PIF que llame a aquél, y elegir en éste qué combinaciones de teclas queremos reservar para el programa. Con eso ALT INTRO queda libre y tiene la misma función que en el QL.

Desde que descubrí esto, a raíz de la llegada del número de septiembre de Quanta, siempre ejecuto SMSQ bajo Windows, aunque suene a herejía. La ventaja es que puedo simultanear actividades del QL con otras para las que habitualmente uso programas de PC (principalmente WP y FileMaker), y no salir ni de Windows ni de SMSQ para nada. Es muy cómodo.

Organización del disco duro

Después de siete años con un disco de arranque y todas las aplicaciones en disquetes, tener un disco duro en el QL supuso un gran cambio. El primer disco que creé lo formateé con 5 megaoctetos, que se quedaron pequeños, así que al cabo de poco lo reformateé con 16 megaoctetos, que de momento es más que suficiente para mi uso de la QXL. Tampoco podría, aunque lo necesitara, utilizar más espacio, pues mi disco duro tiene sólo 125 Mo, y más de una vez he tenido que hacer una severa limpia para vaciarlo de cosas que no uso habitualmente y dejar espacio libre que satisfaga la voracidad de los ficheros temporales de Windows. Probablemente el próximo año compre un segundo disco, dado el precio que están teniendo, y utilice el actual como arranque, para el sistema (me refiero al PC), y el nuevo para los programas y datos.

Para facilitar la instalación y ejecución de los programas, recurrí a un sistema análogo al que uso en el PC: la variable del sistema PATH del PC contiene el directorio de MS-DOS, el de DR-DOS, el de Windows, y un directorio especial llamado BAT donde creo todos los ficheros BAT que ponen en marcha todos los programas que uso o he usado alguna vez. La ventaja de esto es que no hay que cambiar PATH nunca, basta con crear un sencillo cargador para cada programa, y situarlo en el directorio BAT. Este sistema siempre me ha parecido más económico, práctico y cómodo que otros.

Pues, a la hora de diseñar un sistema para el QL, hice lo siguiente:

1) Primero, creé una estructura de directorios en el disco duro:

```
     WIN1_Boot
     WIN1_B_ (ficheros en Basic para botar programas)
     WIN1_L_ (lenguajes)
     Ejemplos:      WIN1_L_SFTH_ (Superforth)
                    WIN1_L_TRB_ (Turbo)
     WIN1_P_ (programas)
     Ejempllos:     WIN1_L_XCH_ (Xchange)
                    WIN1_L_SPC_ (Spectator)
     WIN1_S_ (sistema)
     Ejemplos:      WIN1_S_EXT_ (extensiones varias)
                    WIN1_S_LNG_ (Lightning)
                    WIN1_S_OD_ (Overdrive)
     WIN1_MARCOS_ (mis cosas)
```

2) Convertir el directorio WIN1_B_ en el directorio por defecto para ejecutar programas, y usar el disco RAM1 como unidad temporal para ficheros de ayuda y temporales de todos los programas. Podéis consultar el Boot y los ficheros B_ (botadores guardados en WIN1_B_) que incluyo en este disco de QLíper, para ver cómo está hecho esto. Los ficheros cargadores o botadores son programas en Basic, pero pueden ser ejecutados con EX, lo cual crea una tarea SBASIC nueva y tiene la ventaja de que no elimina el programa Basic de la tarea SBASIC "madre". Cada programa botador mueve los ficheros auxiliares del programa en cuestión a RAM1, pone en marcha éste y después se autodestruye con QUIT.

Para mí el sistema es muy cómodo, porque permite ejecutar cualquier programa en cualquier momento desde Basic, con un simple EX (ni siquiera hace falta incluir para esto la extensión _BAS en la QXL). Espero que a alguien le sirva de algo mi sistema.

No obstante, me he decidido a adquirir QPAC2, pues creo que es imprescindible para sacar rendimiento al sistema (sobre todo la gestión de ficheros y tareas). No sé si podré comprarlo este año, o esperaré un poco más, pero en cualquier caso cuando use QPAC2 utilizaré sus botones (¡si consigo que mi ratón se despierte!).

Dudas sobre la QXL

1) ¿Alguien sabe cómo se organiza la pantalla en la QXL? Quiero decir, dónde se guarda, cómo se sabe en qué resolución estamos? ¿Cómo se puede cargar una pantalla del QL en la QXL, si la resolución es mayor? Quiero decir que, al ser la longitud de las líneas mayor, no se podrá hacer un simple LBYTES en la dirección correspondiente, pues saldría la imagen desplazada un cierto número de octetos en cada línea.

2) ¿Cómo puedo saber si la QXL podrá utilizar una unidad de disco MO (magneto-óptica), interna o externa? ¿Se podrá acceder a una tarjeta SCSI conectada al PC? ¿Y leer un CD-ROM? ¿Y usar una tarjeta de sonido o un módem interno? ¿Alguien sabe algo?

3) El puntero no se mueve con el ratón serie del PC ni empujándolo. Da igual que cambie la configuración del controlador del ratón o la del SMSQ.EXE o lo que sea, lo he intentado todo. Sólo lo puedo probar con la QRAM, que es lo que tengo, pero el PE que uso es el que me vino con la QXL. ¿Hace falta un controlador adicional para la QXL? He leído por ahí que con los ratones serie del QL viene un programa controlador. ¿Hace falta algo así?

4) Si se conecta un segundo disco duro al PC, ¿estará inmediatamente accesible como WIN2? No obstante, me extraña que no haya alguna manera de tener varios discos duros lógicos en la QXL, usando varios ficheros WIN.QXL (WIN1.QXL, WIN2.QXL...) en el mismo disco duro del PC. ¿Alguien ha pensado en esto antes?

5) He oído (mejor dicho leído en Quanta) que Turbo no funciona en la QXL, aunque los programas con él compilados sí. Aún no he hecho una prueba, aunque pienso hacerla para cerciorarme. ¿Alguien tiene experiencia con esto?

6) ¿Para qué sirve QXL_EXT? No he visto nada sobre ello en el manual.

1995 10 15
Marcos Cruz

Acacias 44 (Monteclaro)
ES-28223 Pozuelo de Alarcón
Tel. 91-3513372

FORTH

Superforth 2.0

Hace mucho tiempo que tenía intención de adquirir este programa. El único Forth en el que había trabajado en el QL era el de Computer One, que compré pirateado hace años (mea culpa, kulpo mia!), con su taco de guarrindongas fotocopias y todo. Hice algunas cosas en él, pero nunca me pareció muy cómodo. Para colmo, no podía simultanearse (o sea, multitarear, digo) con nada, aunque creo que las versiones posteriores sí podían.

Por fortuna nunca me decidí a adquirir el SuperForth. Digo por fortuna, porque estaba incluido en la colección de DP cuya propaganda recibí poco después de adquiriri la QXL. Entonces sí que no lo dudé; era una buena ocasión, por 99 GBP (abreviatura de libra, según la nueva norma ISO; pesetas es ESP: siempre una tercera letra añadida a las nuevas abreviaturas de países). Por desgracia, a diferencia del resto de los programas, el SuperForth no incluye el fichero con el manual, así que tuve que hacer un gasto extra (desvergonzadamente abusivo, a mi entender), y pedí una copia impresa del mismo. En el mismo caso se encuentra el Better Basic, al parecer las dos únicas excepciones, silenciadas en la propaganda, donde se dice muy clarito que todos los programas incluyen su documentación en disco en formato de The Editor o Perfection, y que opcionalmente se puede adquirir el manual original ya impreso. De modo que aproveché y compré los dos manuales impresos, por supuesto mandando una fotocopia de su propia propaganda con las frases en cuestión bien marcaditas, a ver si se les caía un poco la cara de verguenza. Cuando recibí los manuales (sin comentario alguno de DP sobre el asunto) me di cuenta de que el del Better Basic no vale para nada, porque el funcionamiento del mismo es evidente... si hubiera probado antes el programa, me habría ahorrado un pico.

Dicho sea entre paréntesis, mientras la atención de Miracle es impecable y sus productos son excelentes, no puedo decir lo mismo de Digital Precision. No sé si esto es impresión mía, pero parecen más interesados en arañar libras de aquí y de allá que en ofrecer un servicio de calidad. Esto lo digo en base a mi propia experiencia. Les he comprado varias cosas en los últimos años, y les he pedido información varias veces sobre sus productos, y nunca me han dejado un buen sabor de boca, al contrario que Miracle, que siempre me han mostrado una actitud más profesional y humana, si puede decirse así también. Una de las cosas que más me exaspera de DP es que cada uno de sus programas es de su padre y de su madre; me dan la impresión de que están escritos y configurados como si fueran lo único que va a funcionar en el QL... por supuesto, casi siempre, en un QL sin expandir, o como mucho con memoria y disquetes.

Algo así me hace sentir también el programa que nos ocupa en este caso: El primer problema con Superforth (SFTH de ahora en adelante) es que está configurado para trabajar en MDV o FLP, pero no en directorios.

Mi copia de SFTH está en el disco duro de la QXL, en la localización WIN1_L_SFTH_ (WIN_L es el subdirectorio de lenguajes, bajo el que está también Turbo, por ejemplo).

Cuando se inicia SFTH con EXEC, lo primero que hace es intentar leer el fichero BLK1 (un fichero de 1 kilooocteto en el formato clásico de los bloques del Forth) del disquete 1, y por supuesto no lo encuentra. Este fichero contiene lo siguiente:

```
( Login block, called by COLD, modify as required )
0 OPEN CON_460x244a0x0   ( the main console window )
2DUP #IN 2! 2DUP #OUT 2!
#DEFAULT CLOSE ' #DEFAULT >BODY 2!
4 MODE 0 PAPER 0 STRIP 7 INK 1 0 CSIZE  ( parameters )
FLP1_
LOAD_FILE FLP1_EDITOR_FTH          ( load the editor )
```

El SFTH puede interpretar ficheros de texto normales y corrientes, con la palabra LOAD_FILE , por ejemplo:

LOAD_FILE flp1_mi_programita_FTH

Esto es un gran avance respecto al sistema tradicional (y, desde mi punto de vista, primitivo y engorroso) del Forth, que consiste en ficheros de texto sin separación de líneas, cuyo contenido se interpreta como una sucesión de bloques de 1024 octetos organizados en 16 líneas de 64 caracteres (lo que se conoce en Forth como una pantalla). Como puede comprenderse, para editar estos ficheros hace falta un editor especial, pues si lo intentamos con un editor de textos normal, al no tener retornos de carro, sólo veremos una línea enorme de texto, de tantos Ko de longitud como pantallas tenga. Podría hacerse si el tratamiento de textos permite visualizar justo 64 caracteres por línea, hace rebosar el texto hacia abajo en vez de hacia la derecha y trabajamos en modo de sobreescritura, pero en cualquier caso es un engorro.

La mayoría de los sistemas Forth incluyen un editor especial para este tipo de ficheros, bien ya incorporado en el vocabulario o, mejor, compilable opcionalmente. SFTH incorpora un programa así en el fichero Editor_FTH, que es cargado por el BLK1. El programa es bastante bueno y cómodo, dentro de lo que cabe.

Aparte, SFTH incluye más ficheros en formato clásico, de BLK2 a BLK9 (una curiosidad de SFTH es que sus ficheros de bloques contienen un solo bloque, o sea 1024 octetos, y el número de éste ha de formar parte del nombre del fichero). Por ejemplo, para cargar e interpretar el bloque 7, es decir el fichero BLK7, hay que hacer: 7 LOAD .

Esto es una extravagancia, pues, ya que permitimos acceso a ficheros de bloque, ¿por qué no que sean de longitud variable y nombre libre? En ese formato es en el que se encuentra mucho logicial antiguo escrito en Forth.

Creo que la respuesta es la siguiente: la posibilidad de utilizar ficheros clásicos de bloques existe sólo para garantizar la compatibilidad con programas antiguos que utilicen este sistema para guardar algún tipo de información. De hecho, dadas las extraordinarias características del QDOS a la hora de leer y grabar ficheros, SFTH no tiene búferes propios para los bloques, ni permite tener en memoria más de un solo bloque, pues no hace falta, ya que el QDOS lo hace.

Lo extraño entonces es que varias pequeñas utilidades vengan en este formato, en lugar del de texto normal. Es más, uno de los bloques que se incluyen en el paquete contiene una pequeña utilidad llamada SAVE_FILE , para convertir ficheros de bloques consecutivos en un solo fichero de texto, pero esta pensada para ficheros de un solo bloque. No es difícil modificarla, pero es extraño que no venga ya preparada para adaptar ficheros de longitud variable. El código de SAVE_FILE es el siguiente:

```
( to compact consecutive blocks into a named file )
2VARIABLE #TEMP
: #SWAP #OUT 2@ #TEMP 2@ #OUT 2! #TEMP 2! ;
: SAVE_FILE             ( first_block last_block --- )
     #OUT 2@ #TEMP 2! 2 OPEN #OUT 2! 1+ SWAP
     DO
       #SWAP I BLOCK #SWAP L/B 0
       DO
```

```
        I C/L * OVER + C/L -TRAILING
        DUP 0> IF TYPE CR ELSE 2DROP THEN
    LOOP DROP LOOP ." END_FILE " CR
    #OUT 2@ CLOSE
    #TEMP 2@ #OUT 2!
;
END_FILE
```

Por cierto, hace algún tiempo estuve escribiendo unas extensiones para F83 (un conocido Forth-83 para MS-DOS) para poder leer ficheros de texto normales, y tuve que escribir, para probarlas, un pequeño programa en Clipper para convertir ficheros de bloques en ficheros de texto.

En fin, el caso es que es mejor olvidarse de los ficheros de bloques, del editor de bloques y de todo eso, y trabajar como el resto del mundo. Pero nos encontramos con la última dificultad, antes mencionada: nada más ponerse en marcha, SFTH intenta cargar el fichero flp1_BLK1 para interpretarlo, a modo de Boot. ¿Qué hacemos con esto?

Primero, para evitar tener que recurrir al editor de bloques cada vez que queramos modificar el dichoso BLK1, lo mejor es sustituir su contenido por una simple instrucción:

```
LOAD_FILE xxxn_BOOT_FTH
```

"xxxn_" representa el dispositivo, como luego veremos.

De este modo, el auténtico fichero de botadura será un fichero de texto normal, editable con The Editor, Spy o lo que más nos guste.

¿Ahora bien, cómo resolver el problema de flp1_?

Una de las utilidades opcionales de SFTH es una palabra, CREATE_DEVICE , que permite añadir dispositivos alternativos, para no estar limitado a FLP y MDV. Por desgracia, no permite dispositivos de más de 5 letras (FDK1_, WIN1_, etc.), así que no nos valdría para WIN1_L_SFTH_, por ejemplo.

Tenemos que recurrir a DEV_USE, no hay más remedio. Así pues, en el boot del SBASIC añadí un tercer DEV_USE:

```
DEV_USE 8, WIN1_L_SFTH_
```

Por tanto, el dispositivo mencionado en BLK1 debe ser DEV8. No obstante, hay que crearlo en SFTH. Para ello basta con cargar la utilidad CREATE_DEVICE , que es la siguiente:

```
( Creates a new default device and enables you to
( switch between it and MDV1_ etc. Use is eg )
(    CREATE_DEVICE FDR1_      then FDR1_ will set a )
( device called FDR1_ as the default )
BASE @ DECIMAL            ( Just in case !! )
: CREATE_DEVICE
     CREATE
        LATEST 5 + @      ( Get characters 3 and 4 )
        DUP 63 AND 48 - , ( Compile device number )
        ,                 ( Compile characters 3 and 4 )
        LATEST 3 + @ ,    ( Compile first 2 characters )
     DOES>  >R R@ @ R@ 2+ @ R> 4 + @ SDV   ;
BASE !                    ( Restore original base )
END_FILE
```

y ordenar:

```
CREATE_DEVICE DEV8_
```

A partir de entonces, DEV8_ es una palabra que activa el dispositivo DEV8. Pero esto solo no basta, pues este dispositivo debe estar disponible desde el primer momento, para que desde él se cargue el fichero BLK1. La solución es crear una copia ejecutable modificada del SFTH que incluya CREATE_DEVICE y la definición de DEV8_ . Esto se consigue con

la palabra TURNKEY, que a su vez es una utilidad opcional que hay que compilar previamente. Una vez compilada podemos hacer, por ejemplo:

```
TURNKEY Saluton!
```

Esto nos muestra en pantalla los datos necesarios para, desde SuperBASIC, hacer un SEXEC que cree un nueva versión ejecutable de SFTH que al ponerse en marcha ejecute la palabra Saluton! . Antes de compilar TURNKEY hay que eliminar de su definición la llamada a DENAME, que borra los nombres del diccionario. Esto está previsto para poder distribuir programas escritos en SFTH sin vender al mismo tiempo un sistema SFTH utilizable. Este es el código que se proporciona:

```
( To generate a standalone EXECable program )
HERE 8 + 32856 ! 1550 ALLOT  ( Block buffer and stack )
HERE 32834 ! HERE 32838 ! LATEST ' COLD 4 + !
FENCE 2+ @ 32840 !   ( VOC-LINK )
: DENAME [ FORTH ] LATEST
    BEGIN DUP @ SWAP DUP 2+ C@ 31 AND 3 + 0 FILL
      DUP 32768 = UNTIL    ;
: TURNKEY ' ['] COLD 62 + ! CLS
." How much data space? " QUERY BL WORD NUMBER DROP CR CR
." Return to SuperBasic with CTRL C, then type:" CR CR
." sexec <filename>," 32776 2@ 4096 0 D- D.
." ," HERE 32768 - 4096 + U. ." ," U. CR CR
." Then remove SUPERFORTH since it cannot be re-entered"
CR CR DENAME SLEEP   ;
END_FILE
```

Y este es el código modificado, con unos sencillos cambios al final:

```
( To generate a standalone EXECable program )
HERE 8 + 32856 ! 1550 ALLOT  ( Block buffer and stack )
HERE 32834 ! HERE 32838 ! LATEST ' COLD 4 + !
FENCE 2+ @ 32840 !   ( VOC-LINK )
: DENAME [ FORTH ] LATEST
    BEGIN DUP @ SWAP DUP 2+ C@ 31 AND 3 + 0 FILL
      DUP 32768 = UNTIL    ;
: (TURNKEY) ' ['] COLD 62 + ! CLS
." How much data space? " QUERY BL WORD NUMBER DROP CR CR
." Return to SuperBasic with CTRL C, then type:" CR CR
." sexec <filename>," 32776 2@ 4096 0 D- D.
." ," HERE 32768 - 4096 + U. ." ," U. CR CR
." Then remove SUPERFORTH since it cannot be re-entered"
CR CR ;
: SAVE_APPLICATION (TURNKEY) DENAME SLEEP ;
: SAVE_SYSTEM (TURNKEY) ;
END_FILE
```

Y este es, por fin, el fichero boot_fth, necesario para crear la nueva versión de SFTH:

```
( boot_fth, llamado por BLK1 )

0 OPEN CON_512x250a0x0
2DUP #IN 2! 2DUP #OUT 2!
#DEFAULT CLOSE ' #DEFAULT >BODY 2!
4 MODE 0 PAPER 0 STRIP 7 INK 1 0 CSIZE 2 1 BORDER
FLP1_

LOAD_FILE flp1_words_fth
LOAD_FILE flp1_create_device_fth
CREATE_DEVICE DEV8_
DEV8_
LOAD_FILE flp1_turnkey_fth
SAVE_SYSTEM COLD
END_FILE
```

Flp1_words_fth contiene una rudimentaria versión de VLIST que se suministra originalmente en uno de los ficheros de bloque, y que ha sido renombrada a WORDS (VLIST era en fig-Forth). No entiendo cómo esta palabra no viene ya incorporada al diccionario, y cómo no es un poco más inteligente... lista ocho palabras por línea, sin tener en cuenta si son más o menos largas, con las inevitables consecuencias. De momento la incluimos tal cual.

No hay que olvidar que, para que todo funcione correctamente, hay que hacer desde BASIC:

DEV_USE 8, FLP1_

Pero, a pesar de tanto empeño, nada funciona: el fichero deja de interpretarse poco antes de cambiar los parámetros de la pantalla. Tras un poco de reflexión, ayudado por dos o tres vistazos al manual, descubrimos que la constante doble #DEFAULT cambia de valor, a modo de truco perverso, cuando FILE_LOAD se pone en marcha. Así que no podemos modificar #DEFAULT dentro de un fichero que está siendo interpretado. Parece que por eso se utiliza un fichero de bloque como botador...

¿Nos rendimos? ¿Utilizamos BLK1 como botador y recurrimos a Editor_FTH cada vez que queramos modificarlo? Al fin y al cabo, no hay que modificarlo muy a menudo.

Para colmo, después de unas pruebas más averiguamos que tampoco podemos alterar la ventana de SFTH desde dentro de un fichero que se está interpretando, porque hace falta modificar la dichosa variable #DEFAULT. Para rizar el rizo, después de interpretar un fichero de texto el control vuelve al teclado, a menos que la llamada se haya producido desde un bloque, lo cual significa que mi plan de llenar el BOOT_FTH con sucesivos FILE_LOAD a otros tantos ficheros _FTH se va a pique...

No queda más remedio que utilizar BLK1 como fichero de botadura, y hacer desde el el cambio de la ventana y las llamadas a los ficheros _FTH que queramos.

Sólo queda un problemilla pendiente: ¿cómo hacer que BLK1 se cargue desde dev8_ en lugar de desde flp1_? El truco de CREATE_DEVICE y SAVE_SYSTEM no me dio resultado... al hacer EX con el resultado, el SFTH se cuelga en negro, ¿qué he hecho mal?, ¿tendrá que ver con el espacio de datos de la tarea?

En fin, harto de tantas pesquisas, decido cortar por lo sano: cargo mi viejo Spy y leo el FORTH83_OBJ. Buscando aquí y allá doy con la cadena "flp1_BLKü" y ni corto ni perezoso la cambio por "dev8_BLKü", a ver qué pasa. Grabo el texto como ejecutable y lo ejecuto... No funciona. No intenta leer flp1_, pero no encuentra el bloque (ni siquiera intenta leer win1_). Cambio dev8 por win1 en Spy y repito la prueba, copiando flp1_BLK1 a win1_BLK1. ¡Funciona! Ahora sí que no entiendo nada.

O sea que no reconoce DEV8 para cargar bloques si no se ha especificado previamente con CREATE_DEVICE , pero sí para cargar ficheros de texto con LOAD_FILE , qué curioso.

En resumidas cuentas, la única solución es cargar a mano CREATE_DEVICE y después usar TURNKEY para crear un nuevo sistema SFTH que tenga ya definido DEV8. ¿Pero por qué los sistemas creados con TURNKEY se cuelgan o suspenden al iniciarse?

¿No será la modificación que hice en el código, añadiendo las palabras SAVE_SYSTEM y SAVE_APPLICATION? Pruebo con el código original, creando un sistema nuevo que llame a una palabra que imprime un mensaje. Al ejecutar la nueva tarea, ésta aparece con JOBS pero es como si no existiera.

Pruebo de nuevo, pero con la palabra de ejemplo que aparece en el manual, que imprime 1000 números y después sale con BYE ... Funciona. ¿Qué pasa aquí? ¿Será el BYE ?

Repito mi palabrita con BYE al final... Parece que la pantalla hace algo, un parpadeo. ¿Por qué antes no se quedaba mi mensaje en la pantalla? No entiendo.

Pruebo de nuevo, esta vez para poner en marcha una palabra que lee e imprime diez pulsaciones del teclado y después sale con BYE. Funciona.

Ahora la primera prueba de fuego: quitar el DENAME del

interior de TURNKEY, para ver si afecta (no tendría por qué): Funciona igual.

A continuación la segunda prueba de fuego: quitar el BYE del final de la palabra: Bien, vuelve al Forth, en realidad sigue ejecutando COLD, que es de donde vino la llamada.

¡Por eso se colgaban las primeras pruebas que hice, porque escribí TURNKEY COLD y se creaba un bucle sin fin!

Después de un día de reflexión, lo vuelvo a intentar. Al utilizar el código original de TURNKEY , tan sólo suprimiendo la llamada a DENAME y utilizando una nueva palabra NOOP (que no hace nada) como enganche, el ejecutable creado se inicia bien (a veces imprime basura al empezar), pero no tiene en el diccionario las palabras nuevas, empezando por TURNKEY , por ejemplo!

Al final, tengo que recurrir a una solución intermedia: crear con Spy una copia de FORTH83_JOB, llamada SFTH_EXE, a la que he cambiado la cadena flp1 por ram1. Crear un cargador en BASIC que copie BLK1 a ram1_ y después ejecute SFTH_EXE, llamado SFTH_BAS:

```
100 REMark Cargador para SuperForth
110 :
120 _1$="dev8_"
130 _2$="ram1_"
140 mueve "blk1"
170 EX _1$ & "sfth_exe"
180 QUIT
190 :
200 DEFine PROCedure mueve(fichero$)
210   COPY_O _1$ & fichero$ TO _2$ & fichero$
220   END DEFine
230 :
240 DEFine PROCedure salva
250   SAVE_O win1_B_sfth_bas
260   END DEFine salva
270 :
```

Este cargador, como el resto de los cargadores análogos que se encuentran en su mismo directorio (win1_B_), puede ser llamado en cualquier momento desde Basic con EX SFTH (ni siquiera hace falta la extensión del fichero en la QXL).

Por último, necesitamos una versión de BLK1 que cargue las extensiones que necesitamos al empezar la sesión en Forth. Por desgracia, si se llama a más de un fichero con LOAD_FILE desde el BLK1, parece que el primero o primeros no se cargan y permanecen abiertos (cada vez entiendo menos). Por ello, el último recurso que nos queda es llamar desde BLK1 a un solo fichero _FTH que contenga todas las extensiones:

```
( Login block, called by COLD, modify as required )
0 OPEN CON_640X455A0X0
2DUP #IN 2! 2DUP #OUT 2! #DEFAULT CLOSE ' #DEFAULT >BODY 2!
4 MODE 0 PAPER 0 STRIP 4 INK 1 0 CSIZE 2 1 BORDER CLS
LOAD_FILE DEV8_BOOT_FTH
```

A su vez, el fichero Boot_FTH no puede contener otros LOAD_FILE, así que es necesario importar en él todas las extensiones y órdenes que nos interesan:

```
( boot_fth, llamado por BLK1 )

LOWER

: CREATE_DEVICE

  ( Creates a new default device and enables you to )
  ( switch between it and MDV1_ etc. Use is eg )
  (    CREATE_DEVICE FDR1_      then FDR1_ will set a )
  ( device called FDR1_ as the default )
```

```
  CREATE
    LATEST 5 + @      ( Get characters 3 and 4 )
    DUP 63 AND 48 - , ( Compile device number )
    ,                 ( Compile characters 3 and 4 )
    LATEST 3 + @ ,    ( Compile first 2 characters )
  DOES>
    >R R@ @ R@ 2+ @ R> 4 + @ SDV
  ;

CREATE_DEVICE DEV8_
DEV8_

: WORDS

  ( Lists all the words in the CURRENT vocabulary )

  CR LATEST
  BEGIN
    8 0          ( 8 names per line )
    DO
      DUP ID. SPACE  ( Print name and space )
      @ DUP 32768 =  ( Link is 32768 at end ... )
      IF LEAVE THEN  ( ... if so, finished )
      LOOP
    CR DUP 32768 =    ( Repeat until 32768 read )
    UNTIL
  DROP
  ;

END_FILE
```

Hubiera sido mejor poder crear un ejecutable de SFTH con las extensiones ya cargadas y el dispositivo DEV8_ ya creado, pero me rindo, lo siento. Salvador, ¿me puedes ayudar?, ¿has tenido el mismo problema?

(A propósito, el fichero fuente del Reversi no venía en la colección de DP, lo cual me ha extrañado mucho, pues viene su listado en el manual. ¿Venía este fichero en el SFTH comprado solo?)

Mientras espero a poder solucionar este asunto de la configuración para el disco duro, así como a hacer algún programa en SFTH para probarlo a fondo, esta es mi lista de observaciones:

Carencias de SFTH:

- Depurador para ejecutar paso a paso los programas.
- Descompilador de palabras, imprescindible para curiosear y comprender el funcionamiento interno del sistema.
- Anidamiento de sucesivos LOAD_FILE (no importa mucho).
- Más información técnica en el manual (incluso hay palabras en SFTH que ni siquiera aparecen mencionadas en él)..
- Un ensamblador.
- Acceso a ficheros clásicos de Forth con varios bloques.

(Excepto el punto relativo al manual, todo lo demás es subsanable por el usuario, aunque sin la citada información técnica es una ardua tarea de ensayo y error en muchos casos, incluso si tenemos fuentes de otros Forths e intentamos adaptarlas.)

Puntos fuertes de SFTH:

- Velocidad de ejecución.
- Fácil acceso a canales y dispositivos del QDOS.
- FILE_LOAD , a pesar de sus limitaciones.
- Multitarea interna, al parecer fácil y eficaz, por los ejemplos del manual, aunque aún no la he probado.
- Permite definir ventanas fuera de los límites 512x256, en

la QXL.

Marcos Cruz
Pozuelo de Alarcón, 1995 10 11

Z88

En busca del Z88 perdido

Me picó el mosquito de adquirir un Z88 al ver el anuncio de Bill Richardson en Quanta. Pensé que, por ese precio (99 GBP), era una herramienta interesante para escribir textos o llevar bases de datos cuando uno está por ahí, y luego tranferirlo al PC o al QL. La verdad es que también influyó un poco en mi decisión el mero capricho de tenerlo, al tratarse de un invento de Sinclair. Somos incorregibles. Porque la verdad es que un PC portátil, más con la QXL instalada, sería muchísimo más útil. Incluso un Mac portátil sería mejor que un PC, aunque en ese caso no podría utilizar el QDOS en él (a no ser que el emulador de QL escrito en C para Mac del que me informó Salvador funcione bien realmente). En cualquier caso, ahora no me puedo permitir un PC o Mac portátil ni loco, así que pensé que el Z88 era una buena herramienta intermedia mientras tanto, que no me serviría para todo, pero sí para lo más fundamental: guardar y procesar información de forma provisional.

Le escribí a Bill pidiéndole que me dijera si tenía algún Z88 español, aunque supuse que no, así cómo si se podía configurar el teclado para usarlo en castellano. Me llamó un día a casa, pero yo no estaba. Le mandé un fax, pero no me respondió. Luego supe por Quanta que había estado bastante enfermo.

A través de las páginas de QLíper supe que Joaquín Gallardo tenía un Z88 comprado en Inglaterra, así que me puse en contacto con él, suponiendo que su Z88 era inglés. Quería saber cómo lo usaba para escribir en castellano. Pero resulta que, aunque comprado en Inglaterra, era un Z88 español, como el de Salvador. Joaquín me dio la dirección de la empresa donde lo adquirió. Les mandé una postal y me mandaron un ejemplar de su boletín de noticias sobre el Z88, una amable carta y una lista de precios. Tenían casi de todo. Les mandé un fax para preguntarles unas dudas, entre ellas si tenían un Z88 español. No tenían. Como solución, me dijeron que adquiriera uno inglés y cambiara el sistema operativo, que está en una ROM o EPROM, copiándolo del de otro usuario de Z88 español. La complicación me pareció tan grande que me desilusioné del todo. Además, el precio de los cartuchos de RAM es muy alto... como para que se te pierda uno (habría que pegarlo con pegamento y todo, por si acaso). Fuera de bromas, lo que más me preocupó es que la máquina está totlamente descatalogada, incluso esta misma empresa, al parecer la última que dispone de material, está deshaciéndose de sus existencias. Incluso venden Z88s usados sin garantía por 40 GBP, como recambio. Esto quiere decir que, si se me estropea el Z88 dentro de unos meses me lo puedo comer con patatas o ponerlo en un marco en la pared...

De momento he dejado de interesarme por la idea, pero no obstante no la he descartado del todo. He aquí unas preguntillas para Salvador y Joaquín:

1) ¿Qué tipo de ROM o EPROM lleva instalada el Z88 español? ¿Sabéis de alguien que pudiera copiarla en una EPROM igual, para cambiarla por la de un Z88 inglés?

2) ¿Qué se hace en el Z88 para acceder a un carácter que no está en el teclado? ¿Existe un método único para todos los programas? ¿Puede configurarse ese método o combinación de teclas? Si hubiera posibilidad de configurar el acceso a los

caracteres castellanos en un Z88 inglés, me lo pensaría de nuevo.

3) ¿Tiene el Z88 las vocales mayúsculas acentuadas? ¿Podéis pasarme de alguna manera, a través de QLíper, una lista de su juego de caracteres completo?

4) ¿Compensa el Z88? Quiero decir, compensan las cosas que se pueden hacer con él y sus virtudes (textos, datos, cálculos, programación; peso, tamaño, resistencia...) sus "excentricidades" (precio, accesorios, pantalla, incompatibilidad...). ¿Lo volveríais a comprar, incluso por la mitad de precio, si se os estropeara del todo ahora?

Marcos Cruz
1995 10 15

Acacias 44 (Monteclaro)
ES-28223 Pozuelo de Alarcón
Tel. 91-3513372

Otros Temas

Historias para no dormir por culpa de mi PC

Cuando tuve que comprar un PC 486 en 1993 para poder utilizar mi QXL, no sabia que tendria que sufrir todo un sinfin de calamidades por culpa de un vendedor que me vendió un equipo con dos años de garantia. Desafortunadamente para mi, el equipo tenia innumerables defectos de fabricación o casi todo su hardware averiado, y la tienda y distribuidor (STRONGER) de mi PC decidieron dejarme tirado. Afortunadamente, he podido solucionar todos los problemas de mi PC sustituyendo a costa de mi bolsillo todas las piezas defectuosas.

Recientemente he conseguido, gracias a un disco CD-ROM distribuido en una revista, un driver para poder trabajar con 256 colores con mi tarjeta gráfica TRIDENT en WINDOWS.

Según un pequeño manual que venia con mi PC 486, mi tarjeta gráfica TRIDENT 1 Mbyte (con driver VESA en ROM) tenia dos discos conteniendo drivers para varios programas, WINDOWS y OS/2. Por lo visto, algún chorizo de turno se quedo con mis 2 discos conteniendo los drivers para mi tarjeta gráfica, y condenándome a no poder nunca utilizar mi tarjeta Trident al 100% de sus posibilidades.

Ahora que puedo trabajar con WINDOWS en 800x600 a 256 colores, los programas de gráficos son más vistosos, y los videos que vienen en los CD-ROM de las revistas se pueden ver a 256 colores (antes a 16 colores dejaban mucho que desear). Pero la VideoBlaster que me vendieron es un modelo anticuado que solamente puede trabajar con driver VGA 640x480 a 16 colores, y para poder utilizarla tengo que cambiar de driver de pantalla. Esto último es muy incómodo, pues cambiar de driver significa reiniciar WINDOWS.

Advertencia: Aunque mi VideoBlaster solamente corre con driver VGA Standard, el programa capturador de imágenes de video, y el famoso microsoft "VIDEO for WINDOW" trabajan con millones de colores por punto en pantalla. Eso es debido a que la VideoBlaster tiene una memoria de video de 2 Mbytes de RAM. Los videos creados utilizando las ventajas de la VideoBlaster solamente funcionan, como es natural, en equipos con VideoBlaster instalada, pero la calidad y velocidad de los fotogramas es en tiempo real.

La última sorpresa no lo es tanto, porque sabia que tenia que pasar, pero solamente pensar que con el dinero que me costó mi PC 486 en 1993, hoy me puedo comprar no uno, sino DOS PCs PENTIUM con hardware MULTIMEDIA. Solamente pensarlo me pone enfermo. Y todo no queda en solamente eso, lo increible es que mi equipo en 1993 era de gama alta, y hoy está descatalogado y muy por debajo de la gama baja actual. Bueno, siempre queda la esperanza de cambiar la placa madre por cuatro perras gordas si fuese algún día realmente necesario.

Salvador Merino, 15/11/1995

```
--------------------------------------------------------------------------------
    Programas contenidos en este disco
--------------------------------------------------------------------------------


**** QXL de Marcos Cruz ***
Boot
marcos_TestMath_bas
B_sfth_bas
B_Editor_bas
B_Spy_bas
B_Perfection_bas
B_Xchange_bas
B_Idis_bas
B_spectator_bas
s_qxl_TecladoQXL_txt
s_qxl_UK_ktb
s_qxl_Esp_ext
s_qxl_GenTabla2_bas
s_qxl_ES_kt
s_tra_impresora_bas
s_tra_dir_tabla_dat
s_tra_test_bas
s_tra_boot_bas
s_tra_tabla_txt

***  Quill a RTF  ***       (Autor:  Pedro Reina)
PresentoQuillRTF_txt
QuillRTF_txt
Quill_h
QuillRTF_c
QuillRTF_bas
QuillRTF_exe
QuillRTF_cnf
QuillRTF.exe

***  EFQL  ***              (Autor: Pedro Reina)
PresentoEFQL_txt
EFQL_txt
EFQL_c
EFQL_bas
EFQL_exe
EFQL_cnf
EFQL.exe
Thor_qls
```